REMARKS

Applicants thank the Examiner for his careful consideration of the subject application. Applicants have carefully considered the Office Action mailed March 17, 2008. Claims 1-17 were rejected and remain pending. Claims 1, 6, 10, and 14 have been amended in this response. The amendments are supported by at least page 67 lines 12-13. Claims 1-9 and 14-17 were rejected under 35 USC 101 and Claims 1-17 were rejected under 35 USC 103. Applicants respectfully request reconsideration based on the amendments and arguments herein.

35 USC 101

The Office Action rejected Claims 1-9 and 14-17 under 35 USC 101 as non-statutory subject matter. Specifically, the Office Action claimed as the specification states "the agent and server and client are preferable embodied as software" and this makes the claims having an agent, server, and client unpatentable under 35 USC 101. Applicants respectfully disagree, traverse this rejection, and assert that the amended claims as written are patentable subject matter.

In *Alappat*, the Court of Appeals for the Federal Circuit enunciated the standard for patentability of software in conjunction with a computer. *In re Alappat. 33 F.3d 152(Fed.Cir. 1994).* Specifically this case states:

> Alappat admits that claim 15 would read on a general purpose computer programmed to carry out the claimed invention, but argues that this alone also does not justify holding claim 15 unpatentable as directed *to nonstatutory subject matter. We agree. **We have held that such programming creates a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software**. In re

> Freeman, 573 F.2d 1237, 1247 n.11, 197 USPQ 464, 472 n.11 (CCPA 1978); In re Noll, 545 F.2d 141, 148, 191 USPQ 721, 726 (CCPA 1976); In re Prater, 415 F.2d at 1403 n.29, 162 USPQ at 549 n.29.

As well the court also stated:

> Under the Board majority's reasoning, a programmed general purpose computer could never be viewed as patentable subject matter under Section 101. This reasoning is without basis in the law. The Supreme Court has never held that a programmed computer may never be entitled to patent protection. Indeed, the Benson court specifically stated that its decision therein did not preclude "a patent for any program servicing a computer." Benson, 409 U.S. at 71. **Consequently, a computer operating pursuant to software may represent patentable subject matter, provided, of course, that the claimed subject matter meets all of the other requirements of Title 35**.

With respect to the instant Application, Applicants note the full quoted sentence from the specification reads "(t)he agent and server and client are preferably embodied as software that operates on a general purpose digital computer and which transform such a computer into a special purpose digital computer for carrying out the method steps of this invention." In light of *In re Alappat*, Applicants assert that this sentence, in and of itself, shows that any software is working in conjunction with hardware and does not, as asserted in the Office Action, lack the necessary physical articles, and is therefore patentable.

Applicants assert that the invention as claimed is patentable subject matter as the software executing on the general purpose general computer transforms the general purpose digital computer into a special purpose digital computer. This special purpose general computer is, per *Alappat*, patentable subject matter. Applicants note that the current claim is directed towards an architecture whereas *Alappat* was directed towards a method. However, Applicants respectfully assert, with respect to the present invention, this is a difference without a distinction.

The software running in conjunction with a physical tangible medium, the general purpose digital computer, transforms it into the special purpose digital computer that *Alappat* states is patentable subject material under 35 USC 103.

Applicants further note, that many other portions of the subject application support this assertion of the software working in conjunction in hardware, for example take Page 15 line 14 "DRM Server 116 is connects via internet network cloud."; and Page 16 line 13-14 "Each Agent and the Server is interconnected by an IP line and the IP network itself;" where a physical connection to via an internet cloud clearly implies any software must be in conjunction with a tangible medium. As well, Page 16 18-22 clearly shows this connection "Each Agent and Server in a preferred embodiment comprises software, such as C++ code stored and running in a digital computer, and which may be included in whole or in part on a computer readable medium such as medium 201." If software is running in a digital computer medium it is associated with this medium. Further, Page 62 line 6 references Figure 22 which illustrates both the agent and the server being associated with physical tangible hardware elements.

In conjunction with numerous references in the specification, Claim 1, 6, 10, and 14 states that the "primary software agent hosted on each of said data storage systems" and that "each failover software agent residing on a host." Applicants assert that this again shows that the software claimed in conjunction with a physical medium, which renders it patentable subject matter. Applicants assert that this position is supported by *In re Alappat. 33 F.3d 152(Fed.Cir. 1994).*

Based on the foregoing, Applicants respectfully submit that the amended, in light of the specification, are statutory subject matter. Applicants therefore respectfully request that these rejections be removed and Claims 1-9 and 14-17 be placed in condition for allowance.

35 USC 103

The Office Action rejected Claims 1-17 under 35 USC 103 as being unpatentable over Sicola et al (US Pre-Grant Publication 2004/0064639), hereinafter Sicola, in view of Mashayekhi et al. (US Patent 6,922,791) hereinafter Mashayekhi. Applicants respectfully assert that Sicola in combination of Mashayekhi does not teach the claimed invention. However, to more clearly claim the current invention Applicants have amended independent Claims 1, 6, 10, and 14. Claims 1, 6, 10, and 14 are architecture, method, and system versions of the current invention.

Applicants assert that Sicola may not be used for a proper 35 USC 103 rejection in combination with Mashayekhi for any of the independent Claims 1, 6, 10, or 14. In *Teleflex v. KSR,* the Supreme Court stated that a proper 35 USC 103 rejection requires the following steps be performed: (1) Determining the scope and content of the prior art; (2) Ascertaining the differences between the claimed invention and the prior art; and (3) Resolving the level of ordinary skill in the pertinent art. *Teleflex Inc. v. KSR Int'l Co.* 127 S.Ct. 1727, 1741, 82 USPQ.2d 1385, 1396 (2007). This three part test has also been reemphasized and promulgated in the Federal Register. *Federal Register*, Vol. 72, No. 195.

Applying the KSR test to determine the scope and the content of the cited art, Applicants first assert that neither Sicola or Mikashayeki, in light of the specification, "disclose software agents," as is claimed. Sicola states his method provides such that "no special host software is required to implement the above features because all replication functionality is totally self contained within each array controller and automatically done without user intervention." (Sicola Col.2 p. 0019). As Sicola states his invention "is totally self contained within each array controller and automatically done without user intervention," Applicants respectfully assert that Sicola can not disclose said "failover software agents, each failover software agent residing on a host" as claimed. Applicants also assert Mikashayeki also does not disclose "software agents" as claimed.

Mashayekhi, states he provides "a failover method and system . . . for a computer system having at least three nodes operating as a cluster." (Mashayekhi abstract). Where that failover method "assign[s][ing] a failover node based on the determined weights of the surviving nodes." (Mashayekhi abstract). This failover method includes "assigning applications running on the failed node to the failover node." (Mashayekhi Col.4 l. 48-49) Mashayekhi states his invention proceeds by "detecting failure of one of the . . . nodes, determining a time of failure . . ., assigning a failover node . . ., assigning applications running on the failed node to the failover node. (Mashayekhi Col.4 l. 44-50) Mashayekhi does not disclose "software agents," a priori, running on hosts, rather Applicants assert it discloses "assigning applications."

Applicants disclose a "primary software agent" "in communication with the data transfer server, the primary software agent configured for performing data transfer operations in response to commands from the data transfer server." Applicants further disclose "failover software agents, each failover software agent residing on a host" where these "software agents . . . [are] within a communication path of a data transfer . . . [and] are designated to take over the data transfer operation in response to one or more data transfer commands." This means that each "software agent" has the capability to be "in communication with the data transfer server" and "performing data transfer operations in response to commands from the data transfer server" and is "residing on a host."

One skilled in the art, considering Sicola and Mashayekhi and the differences therebetween, would not arrive at the current invention. First, Applicants assert no such "software agents" exist in either reference. However, assuming arguendo, that such a reference to "software agents" did occur, there is no reason as to why such they would be located on hosts. Sicola explicitly states his method features requiring "no special host software," rather his functionality is "within each array controller." Mashayeki is concerned with "assigning applications." Neither references would enable one skilled in the art to create software agents" and place them on hosts, which would directly remove one of Sicola's features and would be redundant in light of Mashayeki's "assigning applications."

As well, Applicants respectfully disagree with the Office Action assertion that Mashayekhi discloses wherein said primary software agent further represents a failover software agent for another of said primary software agents in another one of said data store systems. Applicants assert Mikashayeki determines which node is used to recover the failed node by "in the weight determining step, the weight of every one of the surviving nodes is determined," whereas, the current invention differs in that "said protocol . . . [is] determined during configuration of said computer architecture." (Mashayekhi Col.3 l. 50-53). Mikashayeki's determination is, "following failure of a node." (Mashayekhi Col.3 l. 45). Performing his determination of a failover node following failure of a node, Mikashayeki does not disclose that this occurs "during configuration of said computer architecture."

Sicola does not provide the teaching to bridge this gap. Sicola, in general, discloses a system having "a redundant configuration including dual Fibre Channel links interconnecting each of the components of two data storage sites." (Sicola Abstract). Sicola, as noted in the Office Action, does not teach wherein said primary software agent further represents a failover software agent for another of said primary software agents in another one of said data storage systems. Not contemplating that the "primary software agent further represents a failover software agent," Sicola would not teach the method for determining placement of such a software agent.

Further, a person of ordinary skill in the art, considering the art of Sicola and Mikashayeki, would not arrive at the present invention. Mikashayeki explicitly teaches a method to enable "failover policies for a cluster environment having more than two nodes in which the applications running on a failed node may be dynamically allocated." A combination of Sicola and Mikashayeki would, determine failover on the "weight of every one of the surviving nodes." Conversely, the current invention is concerned with "replication paths" not the weight of a node. For, if "another Mobility Agent cannot be found on the same replication path the replication will fail." However, if "there is more than one failover agent in the environment, the order of failover is decided during configuration of the architecture."

Another difference between Mikashayeki and the current invention is that Mikashayeki includes "assigning the applications running on the failed node to the failover node," whereas the current invention "will re-issue the command to another Mobility Agent on the same replication path." (Mashayekhi Col.4 l. 48-49). This means that Mikashayeki "assigns applications" where in the current invention the "software agent" "being remote from the primary software agent," is already present and running. Applicants assert that Sicola does not disclose "failover software agents being remote from the primary software agent . . . each failover software agent residing on a host." Rather, Sicola states that "no special host software is required to implement the above features because all replication functionality is totally self contained within each array controller." (Sicola Col.2 p. 0019).

Applicants respectfully assert that one skilled in the art combining Mikashayeki with Sicola would not arrive at the current invention. First, Sicola, by claiming it is not necessary, teaches away from having "special host software." Second, Mikashayeki would not have need for "failover software agents being remote from the primary software agent" as it is a method to "assigning applications running on the failed node to the failover node." Applicants assert that "assigning applications running on the failed node to the failover node" would remove the need for one skilled in the art to place "failover software agents being remote from the primary software agent."

As neither reference, in combination or in isolation, would, for a person with ordinary skill in the art, make dependant Claims 1, 6, 10, or 14 obvious, these references can not properly form a rejection under 35 USC 103. As a result, Applicants believe the current rejection has been rendered moot and request Claims 1, 6, 10, and 14 be placed in condition for allowance. As Claims 2-5, 7-13 and 15-17 depend on Claims 1, 6, 10, and 14, respectively, and Applicants believe these independent claims are allowable, the dependant claims should be allowable for at least the same reasons.

Conclusion

In view of the foregoing, Applicants believe that the application is in condition for allowance and respectfully request favorable reconsideration.

In the event the Examiner deems personal contact desirable in the disposition of this case, the Examiner is invited to call the undersigned agent at (508) 293-7450.

Please charge all fees occasioned by this submission to Deposit Account No. 05-0889.

Respectfully submitted,

Dated: June 27, 2008

/Joseph D'Angelo/
Joseph D'Angelo (Reg. No. 56,800)
Agent for Applicants
EMC Corporation
Office of General Counsel
176 South Street
Hopkinton, MA 01748
Telephone: (508) 293-7450
Facsimile: (508) 293-7189